In-Situ Data Quality Assurance for Environmental Applications of Wireless Sensor Networks

LEVI B. LARKEY, LUIS M. A. BETTENCOURT, and ARIC A. HAGBERG
Los Alamos National Laboratory

We present a local, distributed algorithm to detect measurement errors and infer missing readings in environmental applications of wireless sensor networks. To bypass issues of non-stationarity in environmental data streams, each sensor-processor node learns statistical distributions of differences between its readings and the readings of its neighbors, as well as differences between its current and previous readings. Scalar physical quantities such as air temperature, soil moisture, and light flux naturally display a great degree of spatiotemporal coherence, which in turn leads to a spectrum of fluctuations between adjacent or consecutive measurements characterized by small variances. This permits stable estimation over a small state space. The estimated distributions of differences are then used in statistical significance tests that exclude rare random errors in measurements at any single sensor, and to infer missing readings. Compared to an alternative method based on Bayesian classifiers, our algorithm is more storage-efficient, learns faster, and is more robust in the face of non-stationary phenomena. Field results from a wireless sensor network (Sensor Web) deployed at Sevilleta National Wildlife Refuge are presented.

Categories and Subject Descriptors: C.2.4 [**Computer-Communication Networks**]: Distributed Systems—*Distributed applications*

General Terms: Algorithms, Measurement

Additional Key Words and Phrases: Wireless sensor networks, data quality assurance

1. INTRODUCTION

Wireless sensor networks consist of multiple sensor-processor nodes that communicate with each other using radio frequencies. Sensor nodes, at present and in the envisioned future, are simple devices that operate within limitations in local memory storage and processing. These constraints, although by no means fundamental, are often the result of the practical considerations of producing devices that are inexpensive, small, and autonomous. In addition, sensor operations, and their communication in particular, are also limited by battery capacity or by the ability to harvest power, e.g. through solar panels.

Networks of distributed sensors are a promising technology because they can sense environments—natural and human made—over an unprecedented range of spatial

Los Alamos Unclassified Report LA-UR-06-1117.

Authors' addresses: L. B. Larkey, Modeling, Algorithms, and Informatics, Computer and Computational Sciences Division, Los Alamos, NM 87545; L. M. A. Bettencourt and A. A. Hagberg, Mathematical Modeling and Analysis, Theoretical Division, Los Alamos, NM 87545

Permission to make digital/hard copy of all or part of this material without fee for personal or classroom use provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.
© 20YY ACM 0000-0000/20YY/0000-0001 $5.00

and temporal scales [Szewczyk et al. 2004; Delin 2005]. However, the vast number of nodes required to cover large areas, over long times, places practical constraints on their individual cost. The drive for low-cost sensors and the need for unattended operation, frequently in harsh environments, requires simple and robust devices. Even the most robust devices, however, are subject to operational faults. Under these circumstances it is crucial that isolated errors in individual components do not jeopardize the operation of the whole network. Thus an important issue for this emerging technology is data quality assurance and robustness of operation under point failures [Elnahrawy and Nath 2003; Bychkovskiy et al. 2003].

A general approach for robustness to point failures is to create partial functional redundancy among nodes in a sensor network. In some distributed sensor applications this emerges naturally because neighboring nodes measure local environments that are temporally and spatially coherent. Then, measurements at adjacent sensors, and at the same sensor over time, although potentially stochastic and non-stationary, display significant amounts of mutual information. Hence data quality can be assured through state co-inference between multiple, partially redundant and correlated readings from neighboring nodes, or from the same node at consecutive times [Estrin et al. 2002].

This paper presents a practical, distributed algorithm for detecting measurement anomalies and estimating missing data in environmental applications of wireless sensor networks. The algorithm has been designed for environmental sensing at the Sevilleta Long Term Ecological Research (LTER) site by a Sensor Web developed at NASA JPL [Delin and Jackson 2000; Delin 2002; 2005; Delin et al. 2005]. Because it is designed to work under current technological constraints on memory and processing, the algorithm is intentionally as simple and easy to implement as we found possible. Processing occurs locally on each node and requires only communication between proximal sensor nodes. Such local, distributed algorithms are desirable for wireless sensor networks, where minimizing the amount of wireless communication is a necessary operational constraint [Meguerdichian et al. 2001].

The remainder of the paper is organized as follows. First, we describe an approach to data quality assurance based on Bayesian classifiers, as proposed in [Elnahrawy and Nath 2004]. Next, we present our solution based on measurement differences and compare the performance and implementation requirements with the Bayesian classifier method. We then test our method on real data streams from a Sensor Web deployed at the Sevilleta LTER site.

2. A BAYESIAN CLASSIFIER METHOD

Bayesian classifier methods are a powerful way to perform sequential estimation, and are therefore a natural formalism for devising learning algorithms in distributed sensor networks. However, the direct implementation of such methods tends to run into the *practical* limitations of these simple devices.

A recent proposal for *context-aware sensors* based on Bayesian classifiers uses statistical correlations between sensor readings to detect outliers and approximate missing readings [Elnahrawy and Nath 2004]. Assume that sensor measurements take values in the interval $[l, u]$, and let $R = \{r_1, ..., r_m\}$ be a disjoint cover of this interval. Each subinterval in R is considered a discrete class, with average precision $(u-l)/m$. Each node has its own classifier, consisting of the state of that node's previous reading, h, and of the measurements from two (indistinguishable) nearby sensors, denoted as $n \in \{(r_i, r_j) \in R \times R, i \leq j\}$. By Bayes' theorem, the conditional probability of a reading r_i, given the previous value h at that sensor and readings n from two nearby neighbors, is

$$P(r_i|h,n) = \frac{P(h,n|r_i)P(r_i)}{P(h,n)}. \tag{1}$$

In addition, to reduce the state space for inference, it is assumed in [Elnahrawy and Nath 2004] that the neighbor's spatial measurements and the temporal information contained in the previous reading are conditionally independent, given the reading of the sensor at the present time, yielding the "Naive Bayes" classifier[1]

$$P(r_i|h,n) = \frac{P(h|r_i)P(n|r_i)P(r_i)}{P(h)P(n)}. \tag{2}$$

The output of the classifier is inferred using the method of maximum *a posteriori* (MAP) estimation [DeGroot 2004], and is given by

$$\arg\max_{r_i \in R} P(r_i|h,n) = \arg\max_{r_i \in R} \frac{P(h|r_i)P(n|r_i)P(r_i)}{P(h)P(n)} = \arg\max_{r_i \in R} P(h|r_i)P(n|r_i)P(r_i), \tag{3}$$

where the denominator can be omitted from the optimization because it does not depend on r_i.

In this approach, a missing reading is approximated by the midpoint of the subinterval returned by the classifier. For outlier detection, the probability of the sensed reading is compared to the probability of the most likely measurement according to the classifier. The sensed reading is determined to be an outlier if the two probabilities differ by more than a user-defined threshold.

This method is exhaustive and powerful in classifying all possible states of the system and learning their likelihood, but runs into practical implementation problems. To see this, consider that each node must learn the parameters of its classifier online. To learn $P(r_i)$, a node keeps a count of the number of times r_i occurs for each of m possible values. To learn $P(h|r_i)$, a node also keeps a count of the number of times h and r_i occur together for each of m^2 possible combinations. Similarly, to estimate $P(n|r_i)$, a node must keep a tally of the number of instances n and r_i occur together, for each of $(m^3+m^2)/2$ possible states. Finally, to compute probabilities for outlier detection, a node learns $P(n)$ online by keeping a count of the number of times n occurs for each of $(m^2+m)/2$ values. $P(h)$ is given by $P(r_i)$ where $r_i = h$ and a node must also keep a count of the total number of instances observed. Thus the total number of states stored is

$$\frac{m^3}{2} + 2m^2 + \frac{3m}{2} + 1. \tag{4}$$

Expression (4) was obtained by considering the measurements of a node relative to *two* neighbors. For $k > 2$ neighbors, the corresponding expression scales with

[1] It is worth noting that these assumptions do not apply to ecological environmental data under most circumstances.

leading exponent $k + 1$.

The size of the state space required for inference is important for two reasons. First, nodes typically have limited storage capacity, which in turn limits precision. Consider the example of covering a range of 100 degrees with 1 degree precision. Then a classifier would have to store 520,151 counts, or roughly 2 megabytes. Secondly, the amount of learning data required to populate the state space is prohibitive in many cases. In the same example at least 5 million learning instances would be necessary for estimation (taken here to be roughly an order of magnitude greater than the size of the state space). To put this into perspective, consider that a node taking a reading every five minutes (e.g., [Delin 2005]) would require about 47 years to populate its state space.

The issue of learning is even more critical in cases involving non-stationary phenomena because the learning rate cannot be slower than the rate at which parameters evolve. For example, in the case of outdoor air temperature, conditions change throughout the day as the sun rises, moves across the sky (e.g., placing sensors in and out of shadows), and sets. In addition, conditions also change with season and from year to year, such that combinations of data that occur frequently during a hot summer will appear rarely during a cold winter, and will differ to the next summer. Thus an important discriminating criterion for any data quality assurance method is that it must operate on a timescale commensurate with that of any non-stationary phenomena being measured. For ecological environmental sensing this time scale is typically a few hours.

3. A METHOD BASED ON DIFFERENCES

We now propose an alternative method for data quality assurance, in which each node learns statistical distributions of *differences* between its readings and those of its neighbor's, and also between its own measurements at different times. Such distributions, together with current measurements from a sensor's neighbors and the sensor's previous reading, are then used to identify anomalous measurements and to infer missing values. Although inspired by the general idea of context-aware sensors, this method is fundamentally different in implementation from the Bayesian approach described in the previous section. Compared to the Bayesian classifier, our method is more storage-efficient, learns faster, and is more robust in the face of non-stationary phenomena.

The crucial assumption required for the method to work is that the observed phenomena are spatiotemporally coherent, so that the measurements at neighboring sensors, and at the same sensor over time, display a large amount of mutual information. This is true of ecological environmental applications, where typical node-to-node spacings are in the range of 100-200 meters or less. Moreover, environmental variables such as air temperature, humidity, light flux, soil temperature, and soil moisture display a substantial amount of temporal correlation. It is assumed below that measurements at different sensors are performed at time intervals which are much smaller than the temporal correlation time. This is a characteristic of Sensor Web measurements, which are synchronous across the entire network. An additional final assumption of the method is that the probability density of the differences has a peak near the mean and tails that taper as differences deviate away

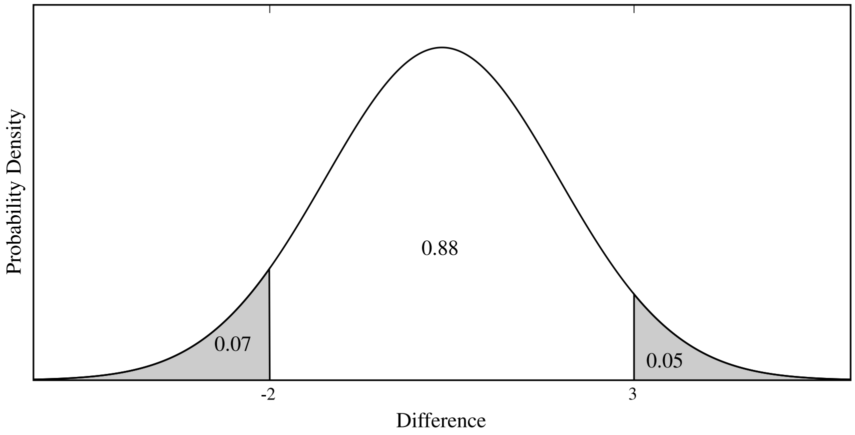

Fig. 1. A statistical probability distribution illustrating the likelihood of observing an extreme difference. In this example, 88% of differences are between -2 and 3, with 7% of differences less than or equal to -2, and 5% greater than or equal to 3.

from it (e.g., see Figure 1). That is, the method assumes that the probability of observing a difference decreases with the distance between that difference and the mean of all observed differences. This is not a strong assumption and could easily be relaxed in more complex circumstances if judged necessary.

Under these circumstances spatial and temporal measurement differences have a more stationary distribution than individual sensor readings. This permits more stable estimation over a much smaller state space. The estimation of differences between sensors placed at different micro-environments, or between those and experimental controls can also capture quantities of direct ecological interest [Collins et al. 2006].

Consider then a node with k neighbors. Let ϕ be the node's reading, ϕ_0 be its previous measurement, and $\phi_i, i = 1, ..., k$, be the readings of its neighbors. At each new measurement the node computes the difference between its current reading and its previous measurement and between its reading and each of its neighbor's $d_i = \phi - \phi_i, i = 0, ..., k$. Given knowledge of the distribution of differences each new observation can be tested for errors. The probability of observing a difference d as or more extreme than d_i is

$$p_i = \min\left[P_i(d \leq d_i), P_i(d \geq d_i)\right] , \tag{5}$$

where the probability P_i is specific to temporal differences or differences with neighbor i. For example, consider the distribution shown in Figure 1, in which 88 percent of differences fall between -2 and 3, with 7 percent of differences less than or equal to -2, and 5 percent greater than or equal to 3. If $d_i = -2$, then $P_i(d \leq -2) = 0.07$ and $P_i(d \geq -2) = 0.93$. Thus $p_i = \min[0.07, 0.93] = 0.07$. Similarly, if $d_i = 3$, then $P_i(d \leq 3) = 0.95$ and $P_i(d \geq 3) = 0.05$. Thus $p_i = \min[0.95, 0.05] = 0.05$.

Each probability distribution P_i is learned from observed differences. There are several ways to implement such an estimation, depending on the degree of prior knowledge. If the distributions are known to belong to a particular class, then learning consists of estimating corresponding functional parameters.

For example, if the distributions are known to be normal, then P_i is defined by its mean and variance, which is estimated in the standard way as

$$\mu_{i,t} = \frac{(t-1)\mu_{i,t-1} + d_{i,t}}{t}\,, \tag{6}$$

$$\sigma^2_{i,t} = \frac{(t-2)\sigma^2_{i,t-1} + (d_{i,t} - \mu_{i,t})^2}{t-1}\,, \tag{7}$$

where t indexes times when differences are observed (for simplicity, assumed here to be synchronous across the network), and $\mu_{i,0} = \sigma^2_{i,0} = \sigma^2_{i,1} = 0$. Thus under parametric estimation a node does not need to store previously observed differences; only the current estimates for the distribution parameters and the number of utilized instances are required. For a normal distribution this is μ_i and σ^2_i for differences in time and differences in space relative to each neighbor, and also t. Thus the total storage required in this case is $2(k+1)$ floating point numbers and an integer, roughly 24 bytes for a node with two neighbors. In addition, the mean and variance can be approximated from as little as 10 observed differences. Discrete distributions, such as Poisson or negative binomial, which may be relevant in many sensing problems, require similar, or smaller, estimation effort and memory storage.

In Equations (6) and (7), the influence of a new difference in approximating μ_i and σ^2_i decreases with the number of previous observations. Therefore, in the case where the distributions are non-stationary, t can be reset at intervals commensurate with the characteristic times of the phenomena under observation. To achieve this in the simplest terms Equations (6) and (7) can be rewritten as

$$\mu_{i,t} = (1-\alpha)\,\mu_{i,t-1} + \alpha d_{i,t} \tag{8}$$

$$\sigma^2_{i,t} = \frac{1-2\alpha}{1-\alpha}\sigma^2_{i,t-1} + \frac{\alpha}{1-\alpha}\left(d_{i,t} - \mu_{i,t}\right)^2, \tag{9}$$

where $\alpha \in (0,1)$ controls the relative influence between the previous and the current observation in updating the parameters. The larger the value of α the higher the weighting of the current observation in the estimation procedure. Note that Equations (8) and (9) are equivalent to Equations (6) and(7) when $\alpha = 1/t$.

By varying α we obtain the best estimator for the distribution parameters under the joint constraints of a limited number of samples and non-stationary data. The limit as $\alpha \to 0$ corresponds to no update of the distribution resulting from the current reading. Even if perfect prior knowledge of the parameters is given at some time, this eventually fails because of the non-stationarity of the phenomenon. As such, the error between actual and predicted data must increase, eventually, as $\alpha \to 0$. On the other extreme, when $\alpha \to 1$, only the current measurement is used in predicting the distribution. This fails because of the standard estimation problem that a small sample of realizations generates imprecise parameter determinations. This reasoning indicates that there is an intermediate value for α that minimizes the error between actual and inferred measurements. We illustrate these features of our scheme in the next section with environmental data from the Sensor Web deployed at the Sevilleta LTER site.

So far we have discussed the case of parametric estimation of known distributions. When the distributions are not known to belong to a particular class, non-parametric estimation is still straightforward, although resulting in slightly larger memory requirements. To do this, we estimate differences over a frequency histogram by dividing the interval of possible differences, $[l_d, u_d]$, into m subintervals. Note that sensor readings are usually subject to device precision and consequently discretization of continuous variables, such as temperature, may not result in further approximation. The average precision, $(u_d - l_d)/m$ achieved in our estimation of differences, is generally much higher than that of the Bayesian classifier, $(u-l)/m$, because $u_d - l_d$ is typically much less than $u - l$. For example, while temperature readings may range from 0 to 100 degrees, differences between temperature readings at neighboring sensors may only vary between -5 and 5 degrees. Thus using 100 subintervals yields an average precision of 0.1 degrees for this method versus 1 degree for the Bayesian classifier.

To approximate P_i, a node keeps a count of the number of times observed differences fall in each subinterval. The probability $P_i(d \leq d_i)$ is the sum of counts for subintervals overlapping $(-\infty, d_i]$, normalized by the sum of all counts. Therefore, in the non-parametric case, a node needs to store $m(k+1)$ integers or roughly $4m(k+1)$ bytes. For example, to cover a range of differences spanning 10 degrees with 1 degree precision, a node with 2 neighbors would have to store 30 states or roughly 120 bytes, whereas the Bayesian classifier would have to store roughly 2 megabytes. In addition, the amount of learning data required to populate the counts is much smaller than for the Bayesian classifier. For example, to cover a range of differences spanning 10 degrees with 1 degree precision would require about 100 observations (roughly an order of magnitude greater than the size of the state space), versus about 5 million learning instances for the Bayesian classifier. In terms of learning time for a node taking a reading every five minutes, this method would require about 9 hours, versus 47 years for the Bayesian classifier. In some cases, a number of measurements commensurate with the size of the state space may suffice, resulting in learning times an order of magnitude below these numbers; however, the ratio between the learning times for each method would be the same.

The probabilities, $p_i, i = 0, ..., k$, are then used to identify anomalous readings. The idea is that anomalous measurements are relatively rare and result from independent point failures that create unusual differences between a node's readings over time and/or between its current measurements and those of its neighbors. The average probability,

$$\overline{p} = \frac{\sum_{i=0}^{k} p_i}{k}, \tag{10}$$

can be compared to a user defined threshold such that readings that are unusually different from previous readings and from readings taken at neighboring nodes are flagged as anomalous. The probabilities are averaged rather than multiplied because they are not independent. Thus, the robustness of the method to false positives due to an anomalous reading at a neighboring node increases with the number of neighbors. The several probabilities p_i can also be weighted based on the uncertainty (e.g. the variance) of the difference distributions to different nodes or the relative degree of spatial versus temporal coherence in the measurements. For example, in sensor networks where nodes are close together, but measurements are infrequent, probabilities associated with spatial differences can be given more weight than those associated with temporal change. Similarly, in sensor networks in which nodes are far apart, but frequent measurements are performed, probabilities associated with temporal differences may be preferred.

A missing reading can be approximated simply by

$$r = \frac{1}{k} \sum_{i=0}^{k} (\phi_i + \mu_i), \tag{11}$$

where ϕ_i is the reading of the ith neighbor and μ_i is the mean difference relative to the ith neighbor, or if $i = 0$, ϕ_0 is the previous reading and μ_0 is the mean difference between the current and previous measurements. A weighted average based on a measure of mutual information between the nodes could also be adopted, but we use the simplest scheme here. In the case where the distribution class is known, μ_i is a stored value. If instead the distribution class is not known, the mean difference can be approximated by its familiar maximal likelihood estimator

$$\mu_i = \frac{1}{m} \sum_{j=1}^{m} c_j m_j, \tag{12}$$

where c_j is the count for the jth subinterval and m_j is the midpoint of the jth subinterval.

4. RESULTS FROM SEVILLETA LTER SITE

In this section, we test our method using ecological data collected by a Sensor Web, developed at NASA/JPL [Delin et al. 2005; Delin 2005], deployed at the Sevilleta LTER site. A Sensor Web is a spatially distributed macro instrument, where every component sensor node (or "pod") shares its readings, at each measurement cycle, with all other pods in the system [Delin and Jackson 2000]. The Sensor Web is designed to maintain synchronicity among all component pods which makes it ideal for the type of correlated statistical analysis proposed in the previous section.

The Sensor Web was initially deployed at the Sevilleta LTER site in 2003 as part of an ongoing effort to measure canopy microclimate effects of three arid land plant species: *Juniperus monosperma* (one-seeded juniper), *Larrea tridentata* (creosote bush), and *Prosopis glandulosa var. torreyana* (honey mesquite) [Collins et al. 2006]. The deployed Sensor Web consists of 14 sensor pods, (see Figure 2) that measure air temperature, humidity, light flux, soil temperature, and soil moisture at a chosen rate (here every five minutes).

The method for inferring missing readings, presented in the previous section, was tested by comparing inferred values to actual measurements. We selected an environmental variable (air temperature), a pod (pod 5), a set of neighbors (pods 8, 9, 11, 12, and 13), and a period of time (the first two days of July, 2005). Pod 5 and its neighbors were chosen to maximize the number of simultaneous measurements during the time period. We used the parametric version of the method [Equations (6)

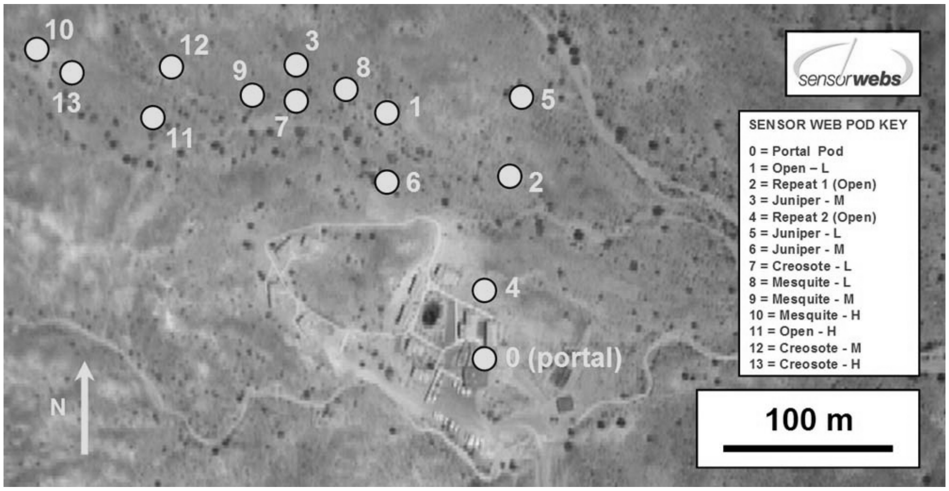

Fig. 2. Aerial photograph showing the Sensor Web layout at the Sevilleta LTER site. Fourteen sensor pods are distributed over a range of a few hundred meters to measure microclimate effects of the surrounding arid land plants. At regular time intervals, the pods transmit data wirelessly to nearby pods. Sensor measurements eventually reach pod 0, where they are recorded.

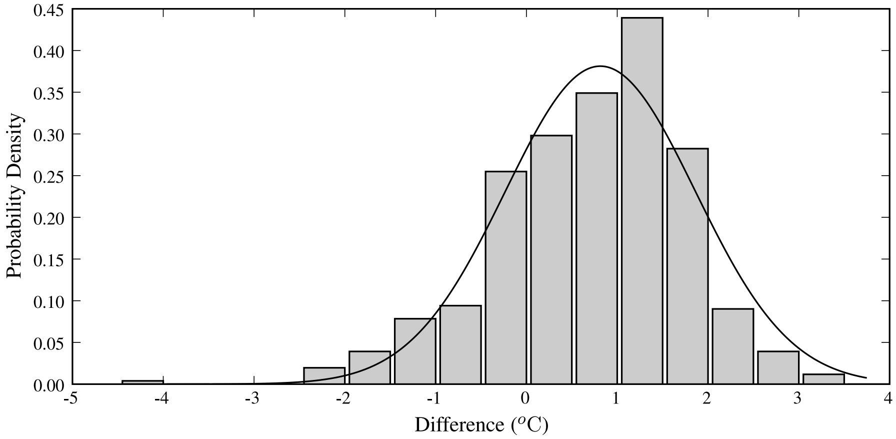

Fig. 3. A histogram of air temperature differences between pod 5 and pod 12 for the first two days of July, 2005. The solid line shows the normal distribution with the same mean and variance.

and (7)] because the distributions of differences are approximately normal (e.g., see Figure 3). Figure 4 shows the inferred and actual readings for pod 5. The average error over the time period was 0.717 degrees Celsius. The spike of 3.3 degrees at 2:57 on July 2 is the result of simultaneous spikes in the readings at pods 8, 9, 11, 12, and 13 of 2.2, 4.0, 5.2, and 4.2 degrees, respectively, while at the same time, the measured air temperature at pod 5 decreased by 0.7 degrees.

Because nodes have different placements, corresponding to distinct micro-climates,

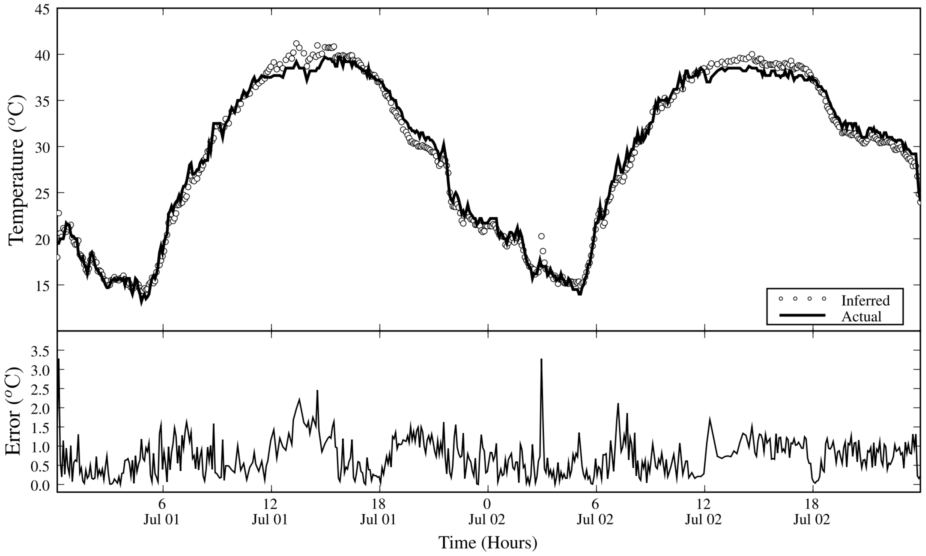

Fig. 4. Actual versus inferred air temperatures at sensor pod 5. The inferred measurements were computed using Eq. (11) and assuming that the distributions of the differences were stationary and approximately normal so that Eqs. (6) and (7) can be used.

the distributions of differences are still somewhat non-stationary. During warmer parts of the day, the more exposed nodes are warmer, but during cooler parts of the day (e.g. at night) the converse is observed (the more exposed nodes are cooler). Under these non-stationary conditions the average measurement error can be reduced by using Equations (8) and (9) with the appropriate value of α that optimizes the learning rate. Figure 5 shows the average error as a function of α. The minimum average error of 0.366 degrees Celsius is achieved for $\alpha = 0.46$. Figure 6 shows the inferred and actual readings for pod 5, using $\alpha = 0.46$.

The difficulty of validating any method for anomaly detection using real data is that there is usually no way to know *a priori* which readings are truly anomalous. To circumvent this problem we created an artificial data set (see Figure 7), composed of a slow amplitude variation of a fast periodic cycle to introduce non-stationary effects, together with random point variations to simulate the effects of sensor errors. With this prescription, time series for three nodes are given by

$$f_1(t) = 11 + 5\sin\left(-\frac{\pi}{2} + \frac{\pi t}{400}\right) + \sin\left(10\left(-\frac{\pi}{2} + \frac{\pi t}{400}\right)\right) + \varphi_1 \tag{13}$$

$$f_2(t) = 9 + 5\sin\left(-\frac{\pi}{2} + \frac{\pi t}{400}\right) + \sin\left(10\left(-\frac{\pi}{2} + \frac{\pi t}{400}\right)\right) + \varphi_2 \tag{14}$$

$$f_3(t) = 6 + 5\sin\left(-\frac{\pi}{2} + \frac{\pi t}{400}\right) + \sin\left(10\left(-\frac{\pi}{2} + \frac{\pi t}{400}\right)\right) + \varphi_3, \tag{15}$$

where $t = 0, ..., 400$ and φ_1, φ_2, and φ_2 are random variables generated from a

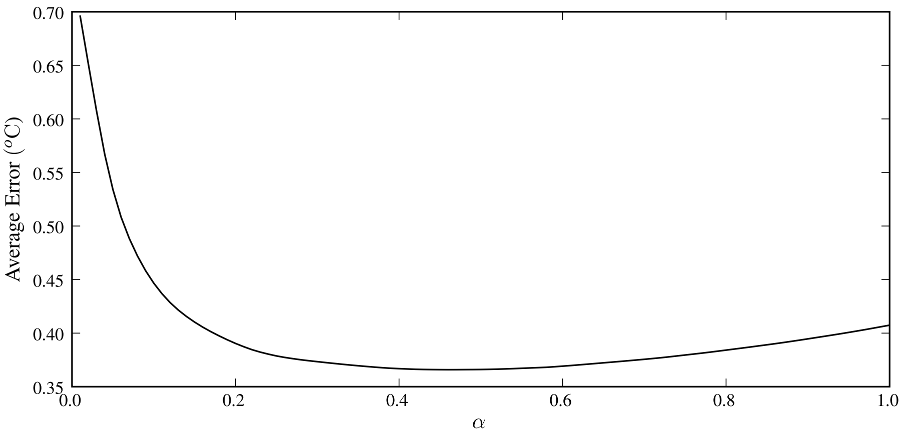

Fig. 5. The average error between the actual data and inferred data as a function of the learning rate, α. The average error is computed using the entire two-day period of measurements. The minimum average error of 0.366 degrees Celsius is obtained for $\alpha = 0.46$.

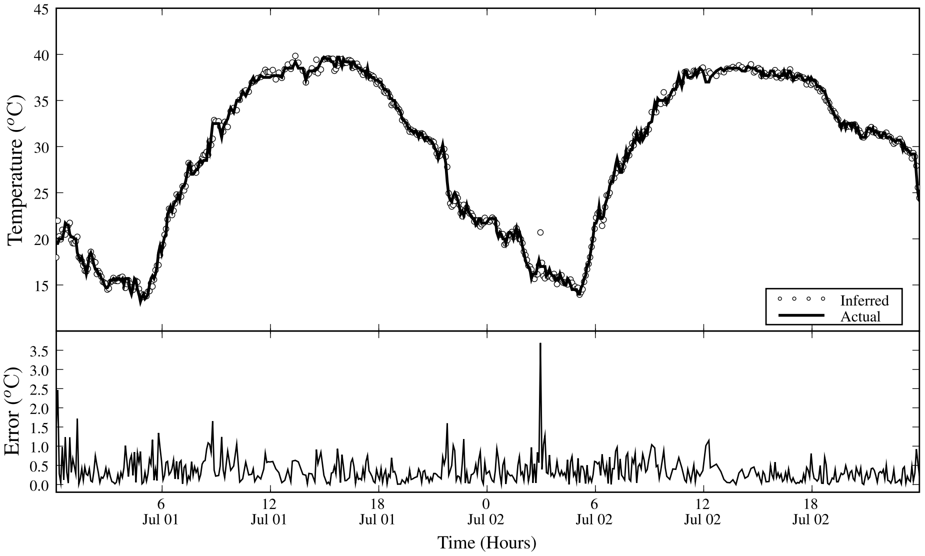

Fig. 6. Actual and inferred air temperatures at sensor pod 5. The inferred measurements were computed using Eq. (11) with estimates of the difference distribution parameters given by Eqs. (8) and (9), with learning rate $\alpha = 0.46$.

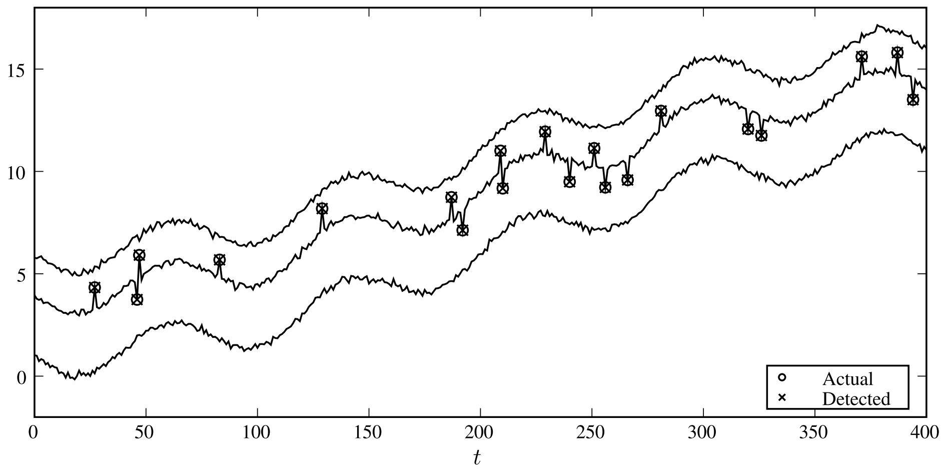

Fig. 7. Actual and detected anomalies for the artificial data sets of Eqs. (13)-(15). Using the parametric version of the method [Eqs. (6) and (7)] and a probability threshold set to $p = 0.005$, all the anomalies are detected correctly. The top, middle, and bottom lines are f_1, f_2, and f_3, respectively.

normal distribution with zero mean and standard deviation $\sigma = 0.1$. In addition, 20 readings from f_2 were randomly selected and perturbed by either -1 or 1. Because the distributions of differences for this example are normal, we used the parametric version of the method [Equations (6) and(7)]. With the probability threshold set to $p = 0.005$, the method displays perfect performance, identifying all the anomalies without any false positives.

5. CONCLUSIONS

We presented a local, distributed algorithm for data quality assurance in wireless sensor networks in which each sensor-processor node learns statistical distributions of differences between its readings and those of its neighbors, as well as for differences between its measurements over time. Each sensor uses these distributions, along with previous and current readings across its neighbors, to identify anomalies and infer missing measurements automatically. The method is intentionally as simple as possible in order to cope with the limited memory and processing capabilities that characterize current sensor network technology. Compared to an alternative method based on Bayesian classifiers, the algorithm proposed here is more storage-efficient, learns faster, and is more robust to non-stationary phenomena. In addition, the storage, processing, and communication requirements are such that it can be implemented in a distributed fashion, on each of the nodes in the network, thus reducing remote communication. Because of these qualities, the algorithm can provide data quality assurance for current generation wireless sensor networks, such as the Sensor Web deployed at the Sevilleta LTER site. In the process of learning distributions of differences for data quality assurance, the algorithm also produces statistics that compare different microclimate environments, to each other and to control experiments, which are of immediate scientific ecological interest.

ACKNOWLEDGMENTS

This work was supported in part by an Intelligence Community Postdoctoral Fellowship to L. B. Larkey. We thank Scott Collins and Renee Brown at Sevilleta, and Kevin Delin at NASA JPL, for enabling and encouraging this work, and Robert Nemzek at LANL for discussions.

The image in Figure 2 was taken by the USDA-ARS Remote Sensing Research Unit, Subtropical Agricultural Research Laboratory, Weslaco, TX using their Aerocommander as the platform for flying a 3-band digital video imaging instrument. Special appreciation goes to M. R. Davis, pilot, USDA-ARS, Weslaco, TX, who flew the Aerocommander 6/3/1997.

REFERENCES

Bychkovskiy, V., Megerian, S., Estrin, D., and Potkonjak, M. 2003. A collaborative approach to in-place sensor calibration. In *Lecture Notes in Computer Science*, F. Zhao and L. Guibas, Eds. Vol. 2634. Springer-Verlag, Berlin, 301–316.

Collins, S. L., Bettencourt, L. M. A., Hagberg, A., Brown, R. F., Moore, D. I., and Delin, K. A. 2006. New opportunities in ecological sensing using wireless sensor networks. Submitted to *Frontiers in Ecology.*

DeGroot, M. H. 2004. *Optimal Statistical Decisions.* Wiley.

Delin, K. A. 2002. The Sensor Web: A macro-instrument for coordinated sensing. *Sensors 2*, 270 – 285.

Delin, K. A. 2005. Sensor Webs in the wild. In *Wireless Sensor Networks: A Systems Perspective*, N. Bulusu and S. Jha, Eds. Artech House.

Delin, K. A. and Jackson, S. P. 2000. Sensor Web for in situ exploration of gaseous biosignatures. In 2000 IEEE Aerospace Conference; Mar 18-25 2000; Big Sky, MT, UnitedStates. *IEEE Aerospace Conference Proceedings 7*, 465 – 472.

Delin, K. A., Jackson, S. P., Johnson, D. W., Burleigh, S. C., Woodrow, R. R., McAuley, J. M., Dohm, J. M., Ip, F., Ferre, T. P. A., Rucker, D. F., and Baker, V. R. 2005. Environmental studies with the Sensor Web: Principles and practice. *Sensors 5*, 103 – 117.

Elnahrawy, E. and Nath, B. 2003. Cleaning and querying noisy sensors. In *Proceedings of the Second ACM International Workshop on Wireless Sensor Networks and Applications.*

Elnahrawy, E. and Nath, B. 2004. Context-aware sensors. In *Lecture Notes in Computer Science*, H. Karl, A. Willig, and A. Wolisz, Eds. Vol. 2920. Springer-Verlag, Berlin, 77–93.

Estrin, D., Culler, D., Pister, K., and Sukhatme, G. 2002. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing 1,* 1 (January), 59–69.

Meguerdichian, S., Slijepcevic, S., Karayan, V., and Potkonjak, M. 2001. Localized algorithms in wireless ad-hoc networks: Location discovery and sensor exposure. In *Proceedings of MobiHOC 2001.* Long Beach, CA, 106–116.

Szewczyk, R., Osterweil, E., Polastre, J., Hamilton, M., Mainwaring, A., and Estrin, D. 2004. Habitat monitoring with sensor networks. *Communications of the ACM 47,* 6 (June), 34–40.

Received Month Year; revised Month Year; accepted Month Year